

Kurs 02422  
Prozeßautomatisierung I

Kurs 21311  
Echtzeitsysteme I

---

Das Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil dieses Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

# Vorwort

In der Prozeßautomatisierung arbeitende Digitalrechner haben die Aufgabe, Programme auszuführen, die mit externen technischen Prozessen assoziiert sind. Auf Anforderung dieser Prozesse hin müssen Erfassung und Auswertung von Prozeßdaten sowie geeignete Reaktionen pünktlich und schritthaltend mit der Prozeßdynamik ausgeführt werden. Dabei steht nicht die Schnelligkeit der Bearbeitung im Vordergrund, sondern die Rechtzeitigkeit der Reaktionen innerhalb vorgegebener und vorhersehbarer Zeitschranken. Die so definierten Echtzeitsysteme sind mithin dadurch charakterisiert, daß die funktionale Korrektheit eines Systems nicht nur vom Resultat einer Berechnung, sondern auch von der Zeit abhängt, wann dieses Resultat produziert wird, womit sie sich grundlegend von anderen Datenverarbeitungssystemen unterscheiden.

Die Begriffe Echtzeit(rechen)systeme, Prozeßdatenverarbeitung bzw. rechnergestützte Prozeßautomatisierung werden wir in diesem Kurs alle synonym für Steuerung und Regelung technischer Prozesse mit programmierbaren elektronischen Systemen verwenden. Unter letzterem Oberbegriff lassen sich speicherprogrammierbare Steuerungen, Industrie-PCs, Prozeßrechner und Prozeßleitsysteme zusammenfassen.

Die technische Entwicklung ist durch das Bestreben gekennzeichnet, technische Vorgänge möglichst kostengünstig durchzuführen. Im Zuge dieser Entwicklung wurden immer mehr selbsttätig arbeitende Einrichtungen zur Meßwerterfassung, Steuerung und Regelung eingesetzt. So entstanden in der Vergangenheit die weitgehend selbständigen Fachgebiete Meß-, Steuerungs- und Regelungstechnik. Mit dem Vordringen der Mikroelektronik werden für alle Automatisierungsaufgaben zunehmend digitale Verfahren eingesetzt, Meß-, Steuer- und Regelgeräte wurden weitgehend digitalisiert. Dies führt nun einerseits dazu, daß die bisher getrennten Fachgebiete wieder enger zusammenwachsen, ja sogar in einigen Fällen durch ihre gemeinsame mikroelektronische Basis miteinander verschmelzen. Andererseits erwächst für die entwickelnden Ingenieure die Notwendigkeit, neben den Verfahren der Meß-, Steuerungs- und Regelungstechnik auch Verfahren der Software- und Rechnertechnik zu beherrschen, um Prozeßautomatisierungssysteme zu konstruieren. Um diese letztgenannten Verfahren und um ihre Anwendung zur Entwicklung von Echtzeitsystemen geht es im vorliegenden Kurs.

Der Kurs gliedert sich in zwei Teile, wovon der erste Fragen des strukturellen, geräte- und programmtechnischen Aufbaus von Prozeßautomatisierungssystemen gewidmet ist. Der zweite Teil befaßt sich dann mit Verfahren zur Durchführung von Prozeßautomatisierungsprojekten unter Verwendung rechnergestützter Werkzeuge.

Im vorliegenden ersten Teil werden zunächst einführende Informationen über Probleme, Strukturen und historische Perspektive industrieller Echtzeitsysteme gegeben und ihre begrifflichen Grundlagen dargelegt. Die nächsten Kurseinheiten befassen sich mit dem strukturellen und gerätetechnischen Aufbau sowie der Prozeßan-  
kopplung rechnergestützter Prozeßautomatisierungssysteme, beschreiben Funktionen und Aufbau von Echtzeitbetriebssystemen unter besonderer Berücksichtigung von Prozessorzuteilungsverfahren und behandeln die Vorgehensweise zur Erstellung von Software für die Prozeßautomatisierung unter Verwendung der sehr leicht verständlichen Echtzeitprogrammiersprache PEARL. Abschließend werden Prinzipien und Aufbau von Echtzeitkommunikationssystemen, insbesondere mit Feldbussen, und Grundlagen der Zuverlässigkeits- und Sicherheitstechnik sowie entsprechende Maßnahmen vorgestellt.

Ziel des Kurses ist es, Kenntnisse und Fähigkeiten zu vermitteln, die die Studierenden in die Lage versetzen, als Ingenieure auf dem Gebiet der Prozeßautomatisierung tätig zu sein. Insbesondere soll der Kurs die nach dem heutigen Stand der Technik anzuwendenden Prinzipien und Verfahren zum gerätetechnischen Aufbau von Prozeßrechnersystemen, die grundsätzlichen Vorgehensweisen bei der durch Echtzeitbetriebssysteme und geeignete Programmiersprachen unterstützten Echtzeitprogrammierung sowie Möglichkeiten zur Kommunikation in verteilten Automatisierungssystemen vorstellen. Darüber hinaus sollen durch Bearbeitung dieses Kurses und insbesondere auch der dazugehörigen Übungsaufgaben die Fähigkeiten erworben werden, Hardware- und Software-Fragen im Zusammenhang mit Prozeßautomatisierungsaufgaben fachgerecht zu klären, die Literatur des Gebietes zu verstehen und sich mit ihrer Hilfe weiterzubilden sowie Automatisierungsprogramme in PEARL zu erstellen und sich schnell in die Handhabung anderer, vergleichbarer Programmiersprachen einzuarbeiten.

Für den Kurs werden Grundkenntnisse des gerätetechnischen Aufbaus und der Programmierung von Datenverarbeitungsanlagen vorausgesetzt. Sehr hilfreich wären Kenntnisse einer höheren, strukturierten Programmiersprache. Darüber hinaus sind gewisse Kenntnisse aus der Physik, der Elektrotechnik und insbesondere der Regelungstechnik zum Verständnis der Beispiele in den Übungen erforderlich.

Der vorliegende Kurs stellt eine völlige Neubearbeitung des gleichnamigen Kurses aus dem Jahre 1985 dar. Weit über die Hälfte seines Umfanges besteht aus neuem Material. Die aus der ersten Version übernommenen Passagen wurden aktualisiert und ergänzt.

Hagen, im Frühjahr 1998

*Prof. Dr. Dr. Wolfgang A. Halang*  
*Prof. em. Dr.-Ing. Dr. h.c. Rudolf J. Lauber*

# Inhaltsverzeichnis

<b>Vorwort</b>	<b>iii</b>
<b>1 Echtzeitrechnungssysteme und industrielle Prozeßautomatisierung</b>	<b>1</b>
1.1 Einleitung . . . . .	2
1.2 Industrielle Prozeßautomatisierungssysteme . . . . .	5
1.3 Beispiel: Ein Chemischer Prozeß . . . . .	9
1.4 Historischer Überblick . . . . .	12
1.5 Grundlegende Konzepte von Echtzeitsystemen . . . . .	15
1.6 Kontinuierliche und diskrete Zeit . . . . .	19
1.7 Ingenieurmäßiger Entwurf harter Echtzeitsysteme . . . . .	22
<b>2 Systemstrukturen</b>	<b>27</b>
2.1 Prozeßautomatisierung in Einzelgerätetechnik . . . . .	28
2.2 Zentralisierte rechnergestützte Automatisierung . . . . .	32
2.3 Kriterien zum Vergleich der Eigenschaften von Automatisierungsstrukturen . . . . .	36
2.4 Redundante Konfigurationen . . . . .	39
2.5 Mehrstufige Automatisierungssysteme . . . . .	47
2.6 Netzgestützte verteilte Systeme . . . . .	50
2.7 Struktur einzelner Prozeßrechner . . . . .	57
<b>3 Prozeßanschaltung</b>	<b>59</b>
3.1 Infrastruktur technischer Prozesse . . . . .	60
3.2 Anschaltung digitaler Signale . . . . .	63
3.3 Unterbrechungswerke . . . . .	70
3.4 Analogausgaben . . . . .	76
3.5 Analogeingaben . . . . .	81
3.6 Programmierbare Prozeßperipherieeinheiten . . . . .	91
3.6.1 Programmierbare Digitalein- und -ausgabe . . . . .	91
3.6.2 Prozeßperipherieeinheiten mit integriertem Mikrocontroller . . . . .	91
3.7 Sensoren und Aktoren . . . . .	94
3.7.1 Sensoren . . . . .	94
3.7.2 Aktoren . . . . .	96
3.8 Störbeeinflussungen auf Prozeßsignalleitungen . . . . .	98
3.8.1 Arten von Störbeeinflussungen . . . . .	98
3.8.2 Gegentaktstörbeeinflussungen . . . . .	99
3.8.3 Gleichtaktstörbeeinflussungen . . . . .	103
3.9 Maßnahmen gegen Störbeeinflussungen . . . . .	103

3.9.1	Maßnahmen zur Verhinderung von Störbeeinflussungen infolge von Erdpotentialdifferenzen . . . . .	104
3.9.2	Maßnahmen zur Sicherung von Prozeßsignaleingabeleitungen gegen elektromagnetische Störfelder . . . . .	105
3.9.3	Erdungsmaßnahmen . . . . .	107
3.9.4	Überspannungsschutz auf Eingängen elektronischer Schaltungen	108
3.10	Serielle Schnittstellen . . . . .	108
<b>4</b>	<b>Echtzeitbetriebssysteme</b>	<b>117</b>
4.1	Anforderungen und Grundbegriffe . . . . .	118
4.1.1	Grundbegriffe . . . . .	118
4.1.2	Forderung nach Rechtzeitigkeit . . . . .	121
4.1.3	Forderung nach Gleichzeitigkeit . . . . .	124
4.1.4	Echtzeittypische Merkmale von Echtzeitbetriebssystemen . . .	124
4.2	Prinzipien der Ablauforganisation . . . . .	127
4.3	Multitasking . . . . .	132
4.4	Synchronisation von und Kommunikation zwischen Rechenprozessen .	140
4.4.1	Allgemeines . . . . .	140
4.4.2	Problematik der Synchronisation . . . . .	143
4.4.3	Semaphore . . . . .	144
4.4.4	Botschaftenaustausch . . . . .	149
4.4.5	Monitore . . . . .	151
4.4.6	Das Rendezvous-Konzept . . . . .	152
4.4.7	Vergleich . . . . .	153
4.5	Behandlung von Zeit und Ereignissen . . . . .	154
4.6	Verteilte Betriebssysteme . . . . .	157
4.7	Zuteilungsverfahren . . . . .	160
4.8	Peripherietreiberprogramme . . . . .	165
<b>5</b>	<b>Echtzeitprogrammierung</b>	<b>169</b>
5.1	Höhere Programmiersprachen in der Echtzeitprogrammierung . . . .	170
5.1.1	Einsatzerwägungen . . . . .	170
5.1.2	Entwicklungsrichtungen der Anwendung maschinenunabhängiger Programmiersprachen im Echtzeitbereich . . . . .	171
5.2	Die Echtzeitprogrammiersprachenfamilie PEARL . . . . .	174
5.2.1	Entstehungsgeschichte . . . . .	174
5.2.2	Entwicklungsziele und Sprachanforderungen . . . . .	175
5.2.3	Übersicht über die wichtigsten Spracheigenschaften . . . . .	176
5.3	Erläuterung ausgewählter Spracheigenschaften von PEARL 90 . . . .	179
5.3.1	Struktur von PEARL-Programmen . . . . .	179
5.3.2	Sprachmittel zur Formulierung algorithmischer Zusammenhänge	183
5.3.3	Sprachmittel zur Ein- und Ausgabe von Prozeßdaten . . . . .	189
5.3.4	Sprachmittel für die Echtzeitprogrammierung . . . . .	195
5.4	Mehrrechner-PEARL für verteilte Prozeßrechensysteme . . . . .	199
5.5	Ein Beispiel für die Anwendung von PEARL . . . . .	207
5.5.1	Aufgabenstellung . . . . .	207
5.5.2	Fachtechnische Lösungskonzeption . . . . .	207
5.5.3	Entwurf der Programmstruktur . . . . .	210

---

5.5.4	Codierung des PEARL-Programmes . . . . .	211
<b>6</b>	<b>Kommunikationsnetze und Feldbusse</b>	<b>223</b>
6.1	Netzarchitekturen . . . . .	224
6.2	Technik lokaler Netze . . . . .	228
6.3	Medienzugangskontrolle in lokalen Netzen . . . . .	232
6.4	Logische Verbindungssteuerung in lokalen Netzen . . . . .	239
6.5	MAP/TOP-Protokoll . . . . .	241
6.6	Feldbusse . . . . .	246
6.7	Konkrete Feldbussysteme . . . . .	252
6.7.1	PROFIBUS . . . . .	253
6.7.2	Interbus-S . . . . .	254
6.7.3	P-NET-Bus . . . . .	254
6.7.4	Bitbus . . . . .	255
6.7.5	CAN-Bus . . . . .	256
6.7.6	DIN-Meßbus . . . . .	256
6.7.7	SMART-Technik . . . . .	256
6.7.8	ASI-Technik . . . . .	257
6.7.9	FIP-Bus . . . . .	257
6.7.10	FAIS-Bus . . . . .	257
6.7.11	SERCOS-Technik . . . . .	258
<b>7</b>	<b>Sicherheit und Zuverlässigkeit von Echtzeitsystemen</b>	<b>259</b>
7.1	Grundlagen . . . . .	260
7.1.1	Begriffsbestimmungen . . . . .	260
7.1.2	Ausfälle und Fehler . . . . .	261
7.1.3	Strategien und Verfahren zur Erhöhung von Sicherheit und Zuverlässigkeit . . . . .	264
7.2	Zuverlässigkeitstechnik . . . . .	264
7.2.1	Zuverlässigkeitskenngrößen . . . . .	264
7.2.2	Zuverlässigkeitsmodelle für Gerätesysteme . . . . .	269
7.2.3	Zuverlässigkeitsmodelle für Programmsysteme . . . . .	271
7.3	Zuverlässigkeitsmaßnahmen . . . . .	273
7.3.1	Klassifizierung von Zuverlässigkeitsmaßnahmen . . . . .	273
7.3.2	Zuverlässigkeitsmaßnahmen bei Gerätesystemen . . . . .	273
7.3.3	Zuverlässigkeitsmaßnahmen bei Programmsystemen . . . . .	275
7.4	Sicherheitstechnik . . . . .	276
7.4.1	Qualitative Definition von Grundbegriffen . . . . .	276
7.4.2	Technische Prozesse mit und ohne sicheren Zustand . . . . .	277
7.4.3	Ereignisfolge bis zum Eintreten eines Unfallschadens . . . . .	278
7.4.4	Quantitative Definition von Sicherheitskenngrößen . . . . .	280
7.4.5	Vorschriften der Sicherheitstechnik . . . . .	281
7.4.6	Sicherheitsnachweis . . . . .	282
7.5	Sicherheitsmaßnahmen . . . . .	282
7.5.1	Klassifizierung von Sicherheitsmaßnahmen bei Echtzeitsystemen	282
7.5.2	Sicherheitsmaßnahmen bei Gerätesystemen . . . . .	284
7.5.3	Sicherheitsmaßnahmen bei Software . . . . .	288
7.5.3.1	Problemstellung . . . . .	288

7.5.3.2	Zuverlässigkeit, Korrektheit und Robustheit von Programmen . . . . .	289
7.5.3.3	Maßnahmen zur Erzielung korrekter Software . . . . .	291
<b>Literaturverzeichnis</b>		<b>305</b>
<b>A Anhang</b>		<b>307</b>